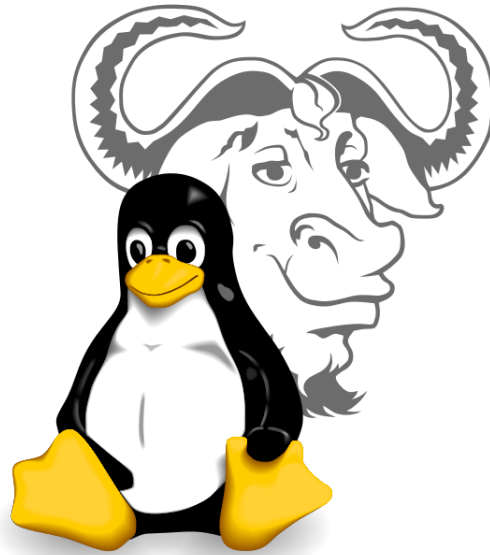


# Obligatorio

## Taller Servidores

### Linux



**Integrantes:** Gonzalez, Federico (245993)  
González, Pablo (238238)

del 2021

10 de Agosto

## Índice

A)	.....	4
Diagrama obligatorio.....		4
Configuración e instalación del controlador:.....		4
Instalación y configuraciones de los servidores CentOS y Ubuntu en VirtualBox.....		10
Instalación de CentOS 8.....		10
Configuramos Interfaces de red NAT y otra Interna.....		15
Chequeo particionamiento con LVM.....		16
Configuración de IP estática .....		16
Configuración nombre de hosts .....		18
Instalacion Ubuntu.....		19
particionamiento LVM en ubuntu.....		21
Configuración de red en ubuntu.....		23
Configuración IP estática en Ubuntu.....		24
B)		
Realizamos Fork del repositorio.....		27
Correccion de playbooks y también agregamos ansible.cfg.....		28
Adaptación de playbooks servidores con UBUNTU o DEBIAN.....		30
C)		
Actualizacion del archivo README.md		

**letra:**

**Prerequisitos:**

Debe tener un equipo bastión (o controlador) con ansible instalado, un usuario para ejecutar los playbooks (recomendamos el usuario ansible) con su clave pública generada.

**A) 30%**

Instalar un servidor CentOS y otro Ubuntu con 1 GB de memoria RAM y 15 GB de Disco (o más), Configure la instalación con el siguiente esquema de particionamiento

- Una partición de 1G para /boot
- usando LVM: un volumen lógico de 5GB para /, 4 GB para /var y 3 GB para /home y 2 GB de Swap.

Debe tener 2 interfaces de red, 1 conectada a NAT y la otra a una red Interna o “Solo Anfitrión” que le permita conectarse al equipo bastión con Ansible.

Agregue un usuario ansible, dele permisos con SUDO sin contraseña. Desde el equipo bastión o controlador, copie la clave pública para poder conectarse al servidor.

**B) 60 % Tareas a realizar con Ansible**

1) Haga un Fork de el repositorio [https://github.com/emverdes/obligatorio\\_2021\\_08](https://github.com/emverdes/obligatorio_2021_08). En ese repositorio se encuentra un rol básico para la configuración de un stack LAMP. Agregue un archivo ansible.cfg y un inventario.

2) Corrija los errores que haya en los playbooks

3) Adapte los playbooks para que se pueda ejecutar también en servidores con UBUNTU o DEBIAN.

**C) 10% Documentación**

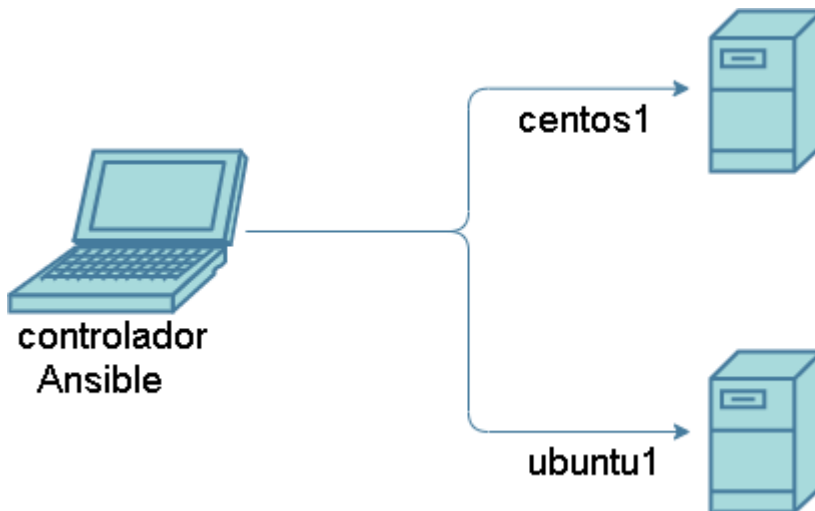
Actualice el archivo README.md explicando los cambios realizados y como utilizar el rol.

Debe documentar cada paso de las tareas realizadas mediante capturas de pantalla. En el caso del playbook debe mostrar la ejecución correcta de todas las tareas.

Agregue la documentación al repositorio. Descargue el repositorio como archivo zip y súbalo a Gestión como entrega.

A)

Diagrama obligatorio:



Instalamos Ubuntu 20.04.2 LTS (GNU/Linux 5.4.72-microsoft-standard-WSL2 el mismo va ser el controlador Ansible y desde ahí vamos administrar los servidores CentOS1 y Ubuntu1. Para poder conectarnos de forma remota a los distintos servidores vamos a instalar y configurar el protocolo SSH.

Configuración e instalaciones del equipo controlador:

```

pablog@PCGonzalez:/home/ansible$ sudo apt-get update
sudo: unable to resolve host controlador: Name or service not known
Get:1 http://security.ubuntu.com/ubuntu focal-security InRelease [114 kB]
Hit:2 http://archive.ubuntu.com/ubuntu focal InRelease
Get:3 http://archive.ubuntu.com/ubuntu focal-updates InRelease [114 kB]
Get:4 http://archive.ubuntu.com/ubuntu focal-backports InRelease [101 kB]
Fetched 328 kB in 5s (71.8 kB/s)
Reading package lists... Done
  
```

```

pablog@PCGonzalez:/home/ansible$ sudo apt install ssh
sudo: unable to resolve host controlador: Name or service not known
Reading package lists... Done
Building dependency tree
Reading state information... Done
ssh is already the newest version (1:8.2p1-4ubuntu0.2).
0 upgraded, 0 newly installed, 0 to remove and 66 not upgraded.
  
```

Instalamos Git

```
pablog@PCGonzalez:/home/ansible$ sudo apt install git
sudo: unable to resolve host controlador: Name or service not known
Reading package lists... Done
Building dependency tree
Reading state information... Done
git is already the newest version (1:2.25.1-1ubuntu3.1).
0 upgraded, 0 newly installed, 0 to remove and 66 not upgraded.
```

```
pablog@PCGonzalez:/home/ansible$ git --version
git version 2.25.1
```

Configuración del ambiente Git:

```
pablog@controlador:/home/ansible$ git config --global user.email "pcgonzalez.aii@gmail.com"
pablog@controlador:/home/ansible$ git config --global color.ui true
pablog@controlador:/home/ansible$ git config --global color.status auto
pablog@controlador:/home/ansible$ git config --global color.branch auto
pablog@controlador:/home/ansible$ git config -l
user.name=pcgGonzalez
user.email=pcgonzalez.aii@gmail.com
color.ui=true
color.status=auto
color.branch=auto
```

Creamos un directorio llamado repo donde vamos a guardar el repositorio del obligatorio  
mkdir repo

```
pablog@controlador:~/repo$ ll
total 12
drwxr-xr-x 3 pablog pablog 4096 Aug  4 21:24 ./
drwxr-xr-x 9 pablog pablog 4096 Aug  5 03:27 ../
drwxr-xr-x 4 pablog pablog 4096 Aug  4 21:24 obligatorio_2021_08/
```

Inicializamos el repo y vemos como se creo el directorio .git

```
pablog@controlador:~/repo$ git init
Initialized empty Git repository in /home/pablog/repo/.git/
```

Dentro del directorio .git podemos ver toda la metadata que está relacionada con nuestro repositorio.

```
pablog@controlador:~/repo$ cd .git
pablog@controlador:~/repo/.git$ ll
total 40
drwxr-xr-x 7 pablog pablog 4096 Aug  5 03:56 ./
drwxr-xr-x 4 pablog pablog 4096 Aug  5 03:56 ../
-rw-r--r-- 1 pablog pablog  23 Aug  5 03:56 HEAD
drwxr-xr-x 2 pablog pablog 4096 Aug  5 03:56 branches/
-rw-r--r-- 1 pablog pablog  92 Aug  5 03:56 config
-rw-r--r-- 1 pablog pablog  73 Aug  5 03:56 description
drwxr-xr-x 2 pablog pablog 4096 Aug  5 03:56 hooks/
drwxr-xr-x 2 pablog pablog 4096 Aug  5 03:56 info/
drwxr-xr-x 4 pablog pablog 4096 Aug  5 03:56 objects/
drwxr-xr-x 4 pablog pablog 4096 Aug  5 03:56 refs/
```

Vamos a clonar el repo que tenemos en GitHub en nuestro equipo local.

```
pablog@controlador:~/repo$ git clone https://github.com/pcgGonzalez/obligatorio_2021_08
```

Con git push subimos los cambios del repositorio local al repositorio remoto

```
pablog@controlador:~/repo$ sudo git push
```

Instalamos Ansible

```
pablog@controlador:/home/ansible$
pablog@controlador:/home/ansible$ sudo apt install ansible
sudo: unable to resolve host controlador: Name or service not known
Reading package lists... Done
Building dependency tree
Reading state information... Done
ansible is already the newest version (2.9.6+dfsg-1).
0 upgraded, 0 newly installed, 0 to remove and 66 not upgraded.
```

Creamos el usuario ansible

**useradd ansible**

creamos el home del usuario

**useradd ansible -m**

Creamos el password del usuario

**passwd ansible**

Instalamos ansible con el comando **sudo apt-get install ansible**

Instalamos Git con el comando **sudo apt-get install git**

## ssh-Keygen

Con el comando ssh-keygen creamos la clave que luego le copiaremos a las VM

```
pablog@PCGonzalez:/#$ su ansible
Password:
$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/ansible/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/ansible/.ssh/id_rsa
Your public key has been saved in /home/ansible/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:APLYWR378y598hkEYreG1yFjLxN58LEmdnN4J1SF4bg ansible@PCGonzalez
The key's randomart image is:
+----[RSA 3072]-----+
| . . . . . +.*==|
| = + . . . X+=o|
| . + . . o X Bo.|
| . o = @ .|
| S + E +|
| = +|
| .. .|
| ..o .o|
| ..+o|
+----[SHA256]-----+
```

Vamos a copiar la clave al servidor remoto para que cuando se ejecute un ssh no tengamos que poner la clave.

En este caso vemos como le creamos la clave para al servidor CentOS que tiene la ip 192.168.43.43

```
$ ssh-copy-id 192.168.43.43
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/ansible/.ssh/id_rsa.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
ansible@192.168.43.43's password:

Number of key(s) added: 1

Now try logging into the machine, with:  "ssh '192.168.43.43'"
and check to make sure that only the key(s) you wanted were added.
```

Ahora hacemos lo mismo para copiar la clave al servidor Ubuntu que tiene la ip 192.168.43.42

```
$ ssh-copy-id 192.168.43.42
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/ansible/.ssh/id_rsa.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
ansible@192.168.43.42's password:

Number of key(s) added: 1

Now try logging into the machine, with:  "ssh '192.168.43.42'"
and check to make sure that only the key(s) you wanted were added.
```

Ahí probamos un ssh manualmente conectando al usuario ansible del servidor CentOS con la ip 192.168.43.43 y podemos ver que se conecta sin pedirnos la clave del usuario ansible.

```
$ ssh ansible@192.168.43.43
Last login: Fri Aug  6 05:52:51 2021 from 192.168.43.38
[ansible@centos1 ~]$ hostname
centos1
[ansible@centos1 ~]$ hostname -I
192.168.43.43 2800:ac:8005:8385:a00:27ff:fe3d:d6de
[ansible@centos1 ~]$ |
```

Ahí probamos un ssh manualmente conectando al usuario ansible del servidor Ubuntu con la ip 192.168.43.43 y podemos ver que se conecta sin pedirnos la clave del usuario ansible.

```
$ ssh ansible@192.168.43.42
Welcome to Ubuntu 20.04.2 LTS (GNU/Linux 5.4.0-80-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Fri 06 Aug 2021 10:03:59 AM UTC

System load:          0.0
Usage of /:           32.1% of 13.71GB
Memory usage:        20%
Swap usage:           0%
Processes:           109
Users logged in:      1
IPv4 address for enp0s3: 192.168.43.42
IPv6 address for enp0s3: 2800:ac:8005:8385:a00:27ff:fe63:8e43

 * Super-optimized for small spaces - read how we shrank the memory
   footprint of MicroK8s to make it the smallest full K8s around.

https://ubuntu.com/blog/microk8s-memory-optimisation

88 updates can be installed immediately.
1 of these updates is a security update.
To see these additional updates run: apt list --upgradable

Last login: Fri Aug  6 09:53:34 2021 from 192.168.43.38
ansible@ubuntu1:~$ hostname
ubuntu1
ansible@ubuntu1:~$ hostname -I
192.168.43.42 2800:ac:8005:8385:a00:27ff:fe63:8e43
```

Con `cat ~/.ssh/id_rsa.pub` vemos el contenido de nuestra clave `id_rsa.pub`

```
pablog@PCGonzalez:~/.ansible$ pwd
/home/pablog/.ansible
pablog@PCGonzalez:~/.ansible$ cat ~/.ssh/id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQGDLPmNdMtT0lMLQvPfi6rmxnL/RWmXZj
/XQmZd/UZPCIX1k8IRxZ/aqHZtbunc/J76IFN8yM6qpRR0bl6ce46aNRikQTpBJjWBudrON
vF6DATEeeTELQS7FA3c1M8+crwITDZAT+nGEdwIamzBC9sBNX0RrqIM7AzjCb8CMfY6yVRY
K13tK4LL7452V/ZRb9cJjSeZOLmJNKZfHFnfJpiLFLGnbZHigkCfFIuoagtxQIHqxR6TOLN
jgCWk7R0g2JDQhXyQeow4UHeiZxt0BDxcsXxwEg4EuABOkMKUkVKZdDX8q93/Mo8LYxiySK
BTCYxCbQzYtI/Lht06ezhq6/n0T7uobRKIgM8CktFMCVks9SDpqOKDWIz7+702EYHfN8c4LK
aQWRc/IBjRXBq72PW/gcOSkPUckD3AzpGNv8SalNk5WvLZBEAToKrubNB7PU0YmoLmZbE9z
ge/YIz6kbv6g2/ujsqsv+68+NBlvgw/kYK1CO/E8tGjv4zq6JFLJ7s= pablog@controla
dor
pablog@PCGonzalez:~/.ansible$
```



Vamos a configurar las direcciones ip con sus nombres en /etc/hosts

```
pablog@controlador:~$ hostname
controlador
pablog@controlador:~$ sudo vi /etc/host
[sudo] password for pablog:
pablog@controlador:~$ sudo vi /etc/hosts
pablog@controlador:~$
```

```
# This file was automatically generated by WSL. To stop auto
automatic generation of this file, add the following entry to /e
tc/wsl.conf:
# [network]
# generateHosts = false
127.0.0.1      localhost
127.0.1.1      PCGonzalez.localdomain  PCGonzalez
192.168.43.38  controlador
192.168.43.42  ubuntu
192.168.43.43  centos
```

Realizamos las pruebas con ping para verificar que hay conexión entre el controlador ansible y los servidores de CentOS y Ubuntu.

```
pablog@PCGonzalez:~$ hostname
controlador
pablog@PCGonzalez:~$ ping centos
PING centos (192.168.43.43) 56(84) bytes of data.
64 bytes from centos (192.168.43.43): icmp_seq=1 ttl=63 time=0.888 ms
^C
--- centos ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.888/0.888/0.888/0.000 ms
pablog@PCGonzalez:~$ ping ubuntu
PING ubuntu (192.168.43.42) 56(84) bytes of data.
64 bytes from ubuntu (192.168.43.42): icmp_seq=1 ttl=63 time=2.31 ms
64 bytes from ubuntu (192.168.43.42): icmp_seq=2 ttl=63 time=1.15 ms
^C
--- ubuntu ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1002ms
rtt min/avg/max/mdev = 1.152/1.731/2.311/0.579 ms
pablog@PCGonzalez:~$ ping controlador
PING controlador (192.168.43.38) 56(84) bytes of data.
64 bytes from controlador (192.168.43.38): icmp_seq=1 ttl=127 time=0.3
```

le vamos a dar permisos de sudoers al usuario ansible.

```
# User privilege specification
root    ALL=(ALL:ALL) ALL
ansible ALL=(ALL) NOPASSWD: ALL
# Members of the admin group may gain root privileges
%admin  ALL=(ALL) ALL
```

## Instalación y configuraciones de los servidores CentOS y Ubuntu en VirtualBox

### Instalación de CentOS 8

Vamos a instalar una VM con centos8, configuramos el nombre del servidor, la carpeta que se va guardar, la versión del S.O y la memoria del mismo.

? X

← Create Virtual Machine

Name and operating system

Name: CentOS

Machine Folder: C:\Users\PCGonzalez\VirtualBox VMs

Type: Linux

Version: Red Hat (64-bit)

Memory size

4 MB 1024 MB 12288 MB

Hard disk

☐ Do not add a virtual hard disk

☒ Create a virtual hard disk now

☐ Use an existing virtual hard disk file

Ubunto20\_taller.vdi (Normal, 10,00 GB)

Guided Mode Create Cancel

En este paso vamos a configurar el tamaño de su disco como nos pide la letra.  
Vamos a cargar la imagen de nuestro S.O CentOS y comenzamos con el proceso de instalación.

? X

← Create Virtual Hard Disk

File location  
C:\Users\PCGonzalez\VirtualBox VMs\CentOS\CentOS.vdi

File size  
4,00 MB 2,00 TB 15 GB

Hard disk file type

- ☒ VDI (VirtualBox Disk Image)
- ☐ VHD (Virtual Hard Disk)
- ☐ VMDK (Virtual Machine Disk)
- ☐ HDD (Parallels Hard Disk)
- ☐ QCOW (QEMU Copy-On-Write)
- ☐ QED (QEMU enhanced disk)

Storage on physical hard disk

- ☒ Dynamically allocated
- ☐ Fixed size
- ☐ Split into files of less than 2GB

Guided Mode Create Cancel

? X

← Select start-up disk

Please select a virtual optical disk file or a physical optical drive containing a disk to start your new virtual machine from.

The disk should be suitable for starting a computer from and should contain the operating system you wish to install on the virtual machine if you want to do that now. The disk will be ejected from the virtual drive automatically next time you switch the virtual machine off, but you can also do this yourself if needed using the Devices menu.

CentOS-8.4.2105-x86\_64-dvd1.iso (9,25 GB)

Start Cancel

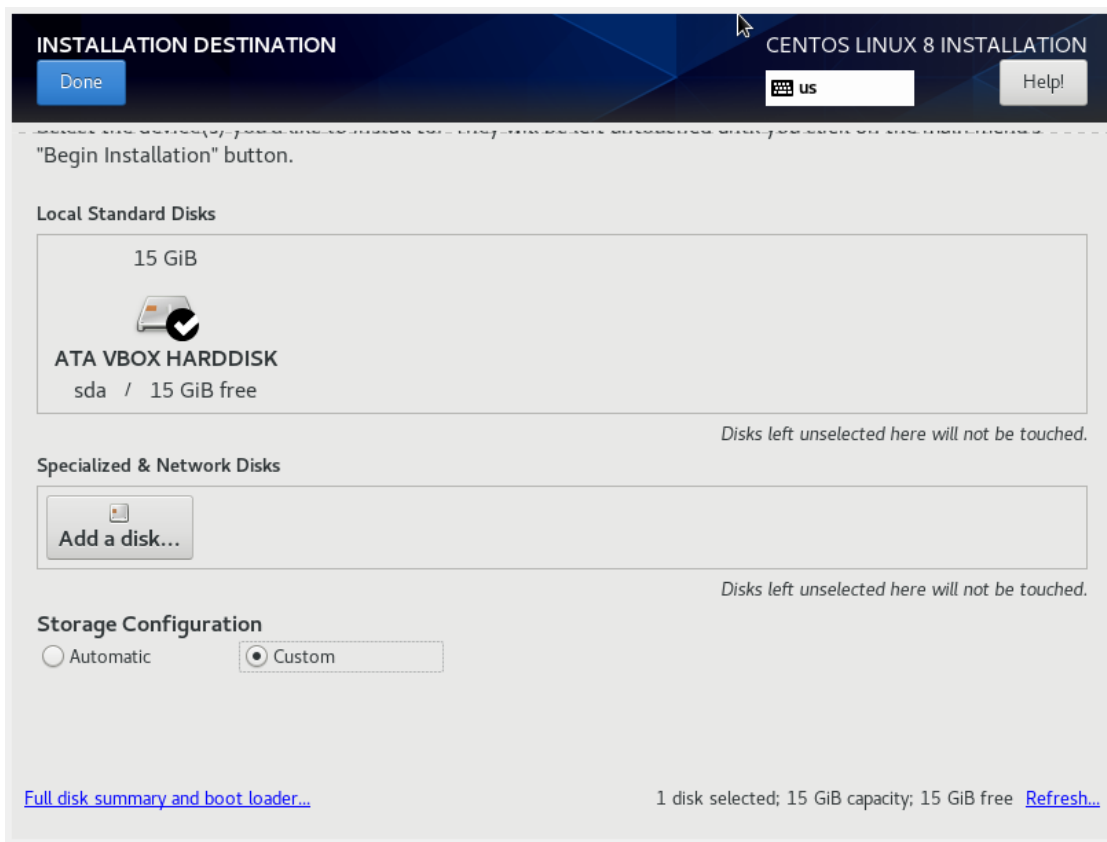
CentOS Linux 8

Install CentOS Linux 8  
Test this media & install CentOS Linux 8

Troubleshooting

Press Tab for full configuration options on menu items.

Ahí vemos el disco que creamos en los pasos anteriores con los 15GB.



Vamos a configurar la instalación con el siguiente esquema de particionamiento LVM como nos pide la letra.

**MANUAL PARTITIONING** CENTOS LINUX 8 INSTALLATION

Done us Help!

---

▼ New CentOS Linux 8 Installation

DATA	Size
/home cl-home	3 GiB
<b>SYSTEM</b>	
/	5 GiB
cl-root	
/var	4 GiB
cl-var	
/boot	1024 MiB
sda1	
swap	2 GiB
cl-swap	

AVAILABLE SPACE: 1023 KiB TOTAL SPACE: 15 GiB

[1 storage device selected](#)

---

**cl-home**

Mount Point: /home Device(s): ATA VBOX HARDDISK (sda)

Desired Capacity: 3 GiB

Device Type: LVM ☐ Encrypt Volume Group: cl (0 B free)

File System: xfs ☒ Reformat

Label: Name: home

Reset All

Vamos a elegir el tipo de instalación : Minimal Install

**SOFTWARE SELECTION** CENTOS LINUX 8 INSTALLATION

Done us Help!

---

**Base Environment**

- ☐ **Server with GUI**  
An integrated, easy-to-manage server with a graphical interface.
- ☐ **Server**  
An integrated, easy-to-manage server.
- ☒ **Minimal Install**  
Basic functionality.
- ☐ **Workstation**  
Workstation is a user-friendly desktop system for laptops and PCs.
- ☐ **Custom Operating System**  
Basic building block for a custom CentOS system.
- ☐ **Virtualization Host**  
Minimal virtualization host.

**Additional software for Selected Environment**

- ☐ **Guest Agents**  
Agents used when running under a hypervisor.
- ☐ **Standard**  
The standard installation of CentOS Linux.
- ☐ **Legacy UNIX Compatibility**  
Compatibility programs for migration from or working with legacy UNIX environments.
- ☐ **Container Management**  
Tools for managing Linux containers
- ☐ **Development Tools**  
A basic development environment.
- ☐ **.NET Core Development**  
Tools to develop .NET and .NET Core applications
- ☐ **Graphical Administration Tools**  
Graphical system administration tools for managing many aspects of a system.
- ☐ **Headless Management**  
Tools for managing the system without an attached graphical console.
- ☐ **Network Servers**  
These packages include network-based servers such as DHCP, Kerberos and NIS.

Crearemos el directorio que utilizaremos.

### ADVANCED USER CONFIGURATION

Home directory:

#### User and Group IDs

☐ Specify a user ID manually:     
☐ Specify a group ID manually:

#### Group Membership

Add user to the following groups:

**Example:** wheel, my-team (1245), project-x (29935)


**Tip:**

You may input a comma-separated list of group names and group IDs here. Groups that do not already exist will be created; specify their GID in parentheses.

Cancel


Save Changes


Comienza la instalación.

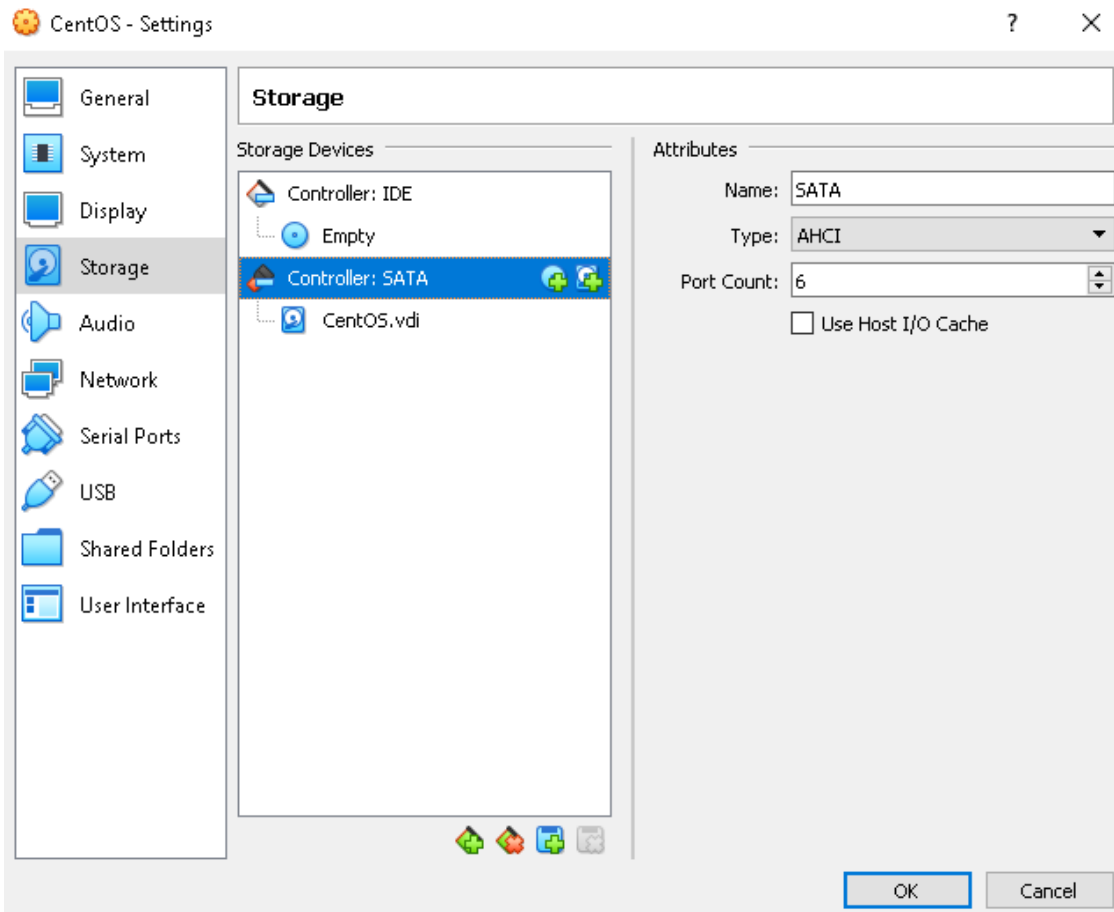
 **CentOS**

INSTALLATION PROGRESS

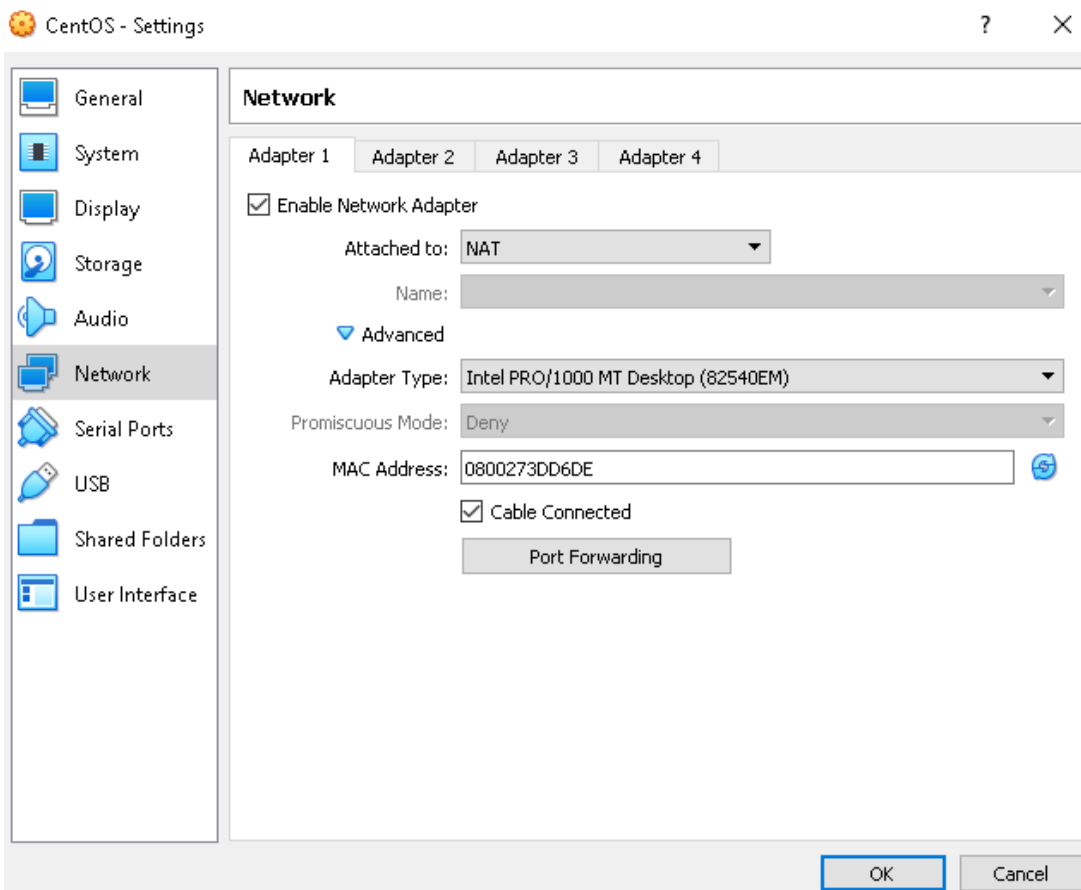
CENTOS LINUX 8 INSTALLATION

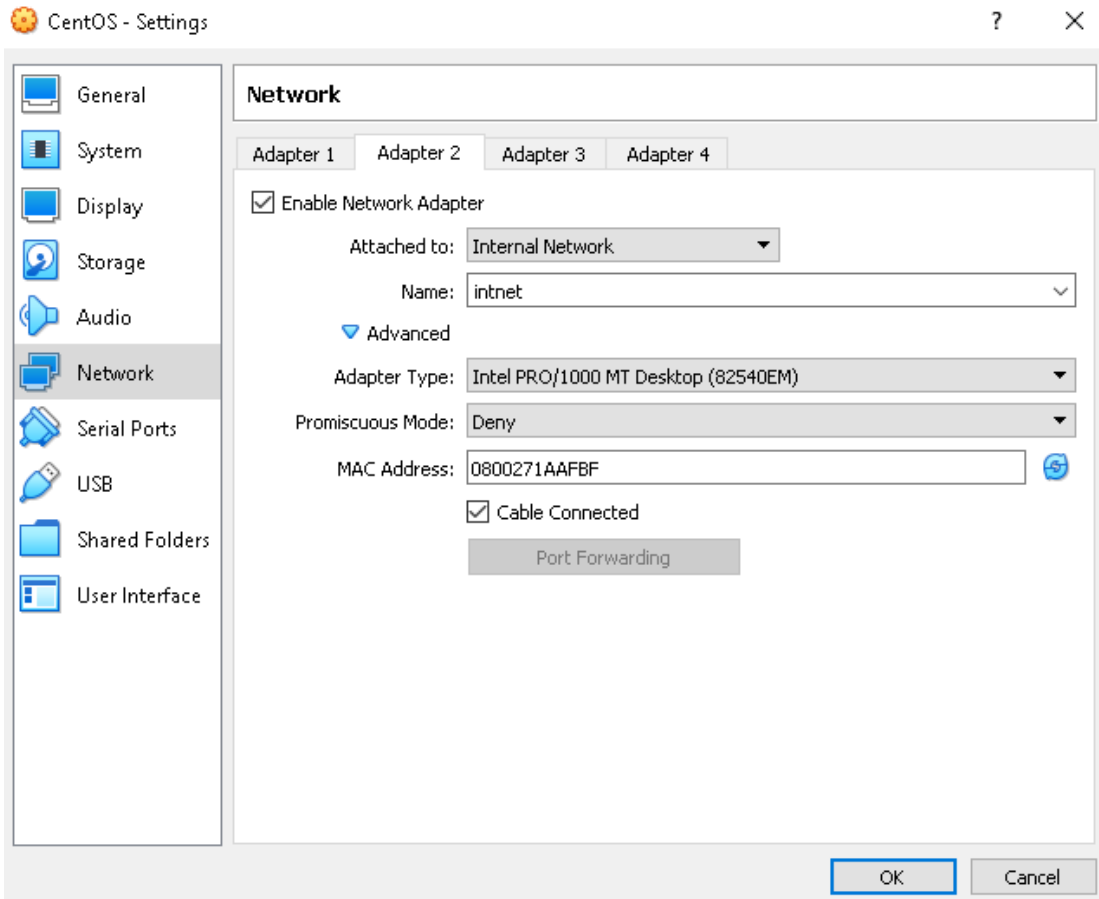


 Creating xfs on /dev/sda1



Configuramos las 2 Interfaces de red 1 a NAT y otra Interna.





Utilizaremos el comando cat para consultar la versión del sistema que se instaló

```
[ansible@centos1 ~]$ cat /etc/os-release
NAME="CentOS Linux"
VERSION="8"
ID="centos"
ID_LIKE="rhel fedora"
VERSION_ID="8"
PLATFORM_ID="platform:el8"
PRETTY_NAME="CentOS Linux 8"
ANSI_COLOR="0;31"
CPE_NAME="cpe:/o:centos:centos:8"
HOME_URL="https://centos.org/"
BUG_REPORT_URL="https://bugs.centos.org/"
CENTOS_MANTISBT_PROJECT="CentOS-8"
CENTOS_MANTISBT_PROJECT_VERSION="8"
```

Creamos el usuario ansible

**useradd ansible**

creamos el home del usuario

**useradd ansible -m**

Creamos el password del usuario

**passwd ansible**

Con pwd vamos a chequear nuestro directorio actual.

```
[ansible@centos1 ~]$ pwd
/home/ansible
[ansible@centos1 ~]$
```



Con el comando LSBLK vamos a ver en pantalla toda la información asociada a las particiones que realizamos con LVM.

```
[ansible@centos1 ~]$ lsblk
NAME        MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
sda          8:0    0   15G  0 disk
├─sda1       8:1    0    1G  0 part /boot
├─sda2       8:2    0   14G  0 part
│   └─cl-root 253:0    0    5G  0 lvm /
│       └─cl-swap 253:1    0    2G  0 lvm [SWAP]
│           └─cl-home 253:2    0    3G  0 lvm /home
│               └─cl-var 253:3    0    4G  0 lvm /var
sr0         11:0    1 1024M  0 rom
```

```
/home/ansible
[ansible@centos1 ~]$ df -h
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        387M   0   387M   0% /dev
tmpfs           405M   0   405M   0% /dev/shm
tmpfs           405M  5.6M   400M   2% /run
tmpfs           405M   0   405M   0% /sys/fs/cgroup
/dev/mapper/cl-root 5.0G  1.7G   3.4G  33% /
/dev/sda1       1014M  256M   759M  26% /boot
/dev/mapper/cl-var 4.0G  319M   3.7G   8% /var
/dev/mapper/cl-home 3.0G   54M   3.0G   2% /home
tmpfs           81M    0    81M   0% /run/user/1000
```

Vamos a configurar IP estática para el servidor CentOS.

entramos a esta ruta **/etc/sysconfig/network-scripts/ifcfg-enp0s3**

```
[ansible@centos1 ~]$ sudo vi /etc/sysconfig/network-scripts/ifcfg-enp0s3
[sudol] password for ansible:
```

```
TYPE=Ethernet
PROXY_METHOD=none
BROWSER_ONLY=no
BOOTPROTO=none
DEFROUTE=yes
IPV4_FAILURE_FATAL=no
IPV6INIT=yes
IPV6_AUTOCONF=yes
IPV6_DEFROUTE=yes
IPV6_FAILURE_FATAL=no
NAME=enp0s3
UUID=1bf68173-16d4-4705-9a98-3c04d6a6b7ca
DEVICE=enp0s3
ONBOOT=yes
IPADDR=192.168.43.43
PREFIX=24
GATEWAY=192.168.43.1
DNS1=8.8.8.8
DNS2=8.8.4.4
```

Guardamos los cambios y reiniciamos el servicio `#systemctl restart network.service`

Chequeamos las IP e Interfaces.

```
[ansible@centos1 ~]$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:3d:d6:de brd ff:ff:ff:ff:ff:ff
    inet 192.168.43.43/24 brd 192.168.43.255 scope global noprefixroute enp0s3
        valid_lft forever preferred_lft forever
    inet6 2800:ac:8004:a880:a00:27ff:fe3d:d6de/64 scope global dynamic noprefixroute
        valid_lft 3309sec preferred_lft 3309sec
    inet6 fe80::a00:27ff:fe3d:d6de/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:3d:d6:de brd ff:ff:ff:ff:ff:ff
    inet 192.168.43.42/24 brd 192.168.43.255 scope global noprefixroute enp0s8
        valid_lft forever preferred_lft forever
    inet6 2800:ac:8004:a880:a00:27ff:fe3d:d6de/64 scope global dynamic noprefixroute
        valid_lft 3309sec preferred_lft 3309sec
    inet6 fe80::a00:27ff:fe3d:d6de/64 scope link noprefixroute
        valid_lft forever preferred_lft forever

[ansible@centos1 ~]$ hostname
centos1
[ansible@centos1 ~]$ hostname -I
192.168.43.43 2800:ac:8004:a880:a00:27ff:fe3d:d6de
```

Vamos a configurar los nombre de hosts en el servidor Centos

```
[ansible@centos1 ~]$ sudo vi /etc/hosts
[sudo] password for ansible:
```

Ingresamos las direcciones IP de los servidores y su nombre.

```
| 127.0.0.1 localhost localhost.localdomain localhost4
localhost4.localdomain4
::1 localhost localhost.localdomain localhost6 local
host6.localdomain6
#192.168.0.143 centos
#192.168.0.142 ubuntu
#192.168.0.141 controlador

192.168.43.38 controlador
192.168.43.42 ubuntu
192.168.43.43 centos
~
~
~
```

Ahora probamos que al realizar ping desde el servidor CentOS al servidor controlador y al servidor ubuntu con el nombre asociado a su ip va a responder.

```
[ansible@centos1 ~]$ hostname
centos1
[ansible@centos1 ~]$ ping controlador
PING controlador (192.168.43.38) 56(84) bytes of data.
64 bytes from controlador (192.168.43.38): icmp_seq=1 ttl=128 time=1.2
8 ms
^C
--- controlador ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 1.277/1.277/1.277/0.000 ms
[ansible@centos1 ~]$ ping ubuntu
PING ubuntu (192.168.43.42) 56(84) bytes of data.
64 bytes from ubuntu (192.168.43.42): icmp_seq=1 ttl=64 time=3.02 ms
^C
--- ubuntu ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 3.017/3.017/3.017/0.000 ms
[ansible@centos1 ~]$ |
```

## Instalación Ubuntu


Vamos a instalar una VM con Ubuntu, configuramos el nombre del servidor, la carpeta que se va guardar, la versión del S.O y la memoria del mismo.

## ← Create Virtual Machine

Name and operating system

Name:

Machine Folder:

Type:  

Version:

Memory size

1024 MB

4 MB 12288 MB

Hard disk

☐ Do not add a virtual hard disk

☒ Create a virtual hard disk now

☐ Use an existing virtual hard disk file

Guided Mode **Create** Cancel

Vamos a cargar la imagen de nuestro S.O Ubuntu y comenzamos con el proceso de instalación

## Create Virtual Hard Disk

File location

File size

15.0 GB

4,00 MB 2,00 TB

Hard disk file type

☒ **VDI (VirtualBox Disk Image)**

☐ VHD (Virtual Hard Disk)

☐ VMDK (Virtual Machine Disk)

☐ HDD (Parallels Hard Disk)

☐ QCOW (QEMU Copy-On-Write)

☐ QED (QEMU enhanced disk)

Storage on physical hard disk

☒ Dynamically allocated

☐ Fixed size

☐ Split into files of less than 2GB

Guided Mode **Create** Cancel

## ← Select start-up disk

Please select a virtual optical disk file or a physical optical drive containing a disk to start your new virtual machine from.

The disk should be suitable for starting a computer from and should contain the operating system you wish to install on the virtual machine if you want to do that now. The disk will be ejected from the virtual drive automatically next time you switch the virtual machine off, but you can also do this yourself if needed using the Devices menu.

**Start** Cancel

Ubuntu [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Willkommen! Bienvenue! Welcome! Добро пожаловать! Welkom!

[ Help ]

Use UP, DOWN and ENTER keys to select your language.

```
[ Asturianu                ▶ ]
[ Bahasa Indonesia         ▶ ]
[ Català                   ▶ ]
[ Deutsch                  ▶ ]
[ English                   ▶ ]
[ English (UK)              ▶ ]
[ Español                  ▶ ]
[ Français                 ▶ ]
[ Hrvatski                 ▶ ]
[ Latviski                 ▶ ]
[ Lietuviškai              ▶ ]
[ Magyar                   ▶ ]
[ Nederlands               ▶ ]
[ Norsk bokmål             ▶ ]
[ Polski                   ▶ ]
[ Suomi                    ▶ ]
[ Svenska                  ▶ ]
[ Čeština                  ▶ ]
[ Ελληνικά                 ▶ ]
[ Беларуская               ▶ ]
[ Русский                  ▶ ]
[ Српски                   ▶ ]
[ Українська               ▶ ]
```

Elegimos para configurar manualmente el particionamiento LVM.

Ubuntu [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Guided storage configuration

[ Help ]

Configure a guided storage layout, or create a custom one:

( ) Use an entire disk

[ VBOX\_HARDDISK\_VBcefe881b-7f19bbbe local disk 15.000G ▼ ]

[X] Set up this disk as an LVM group

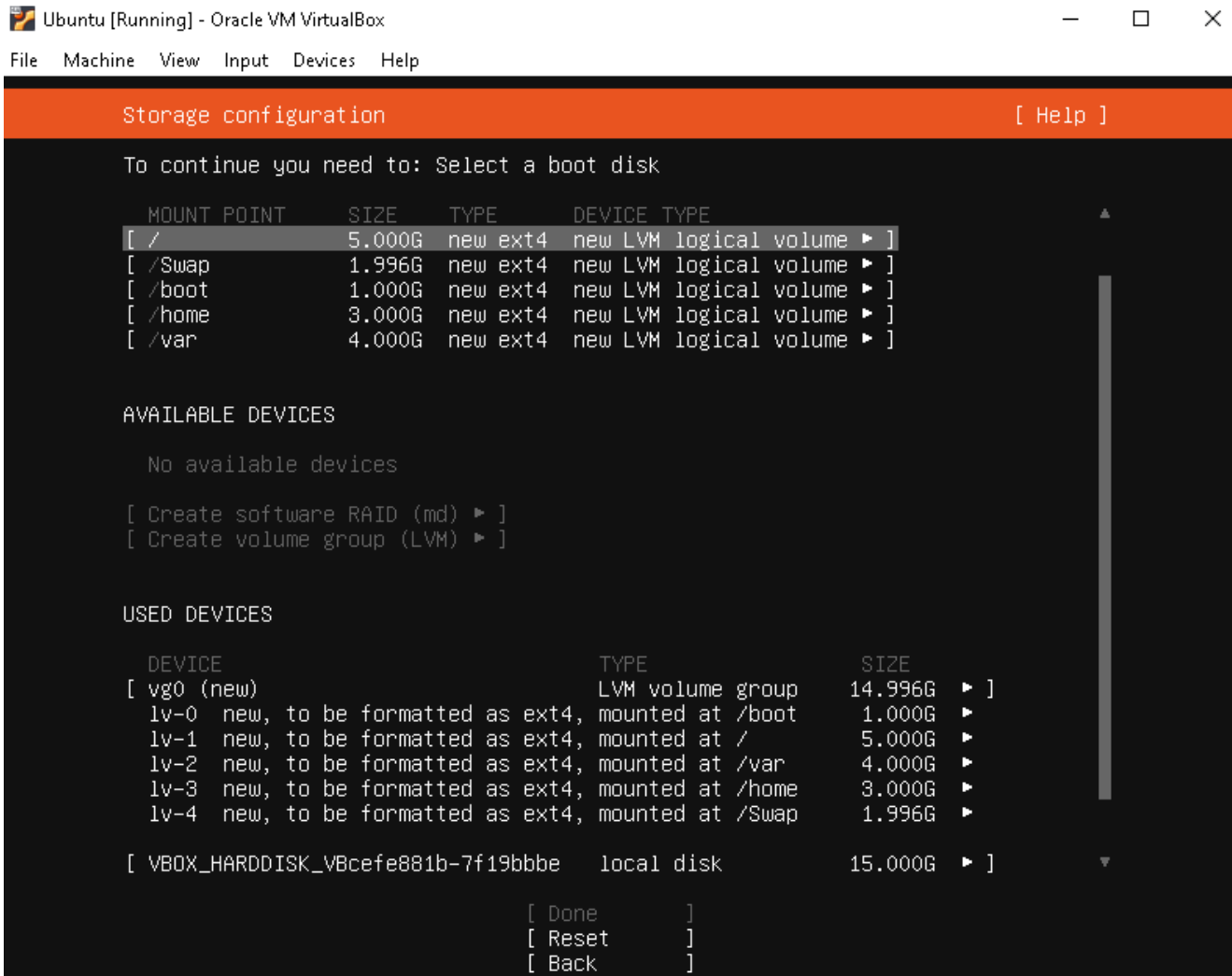
[ ] Encrypt the LVM group with LUKS

Passphrase:

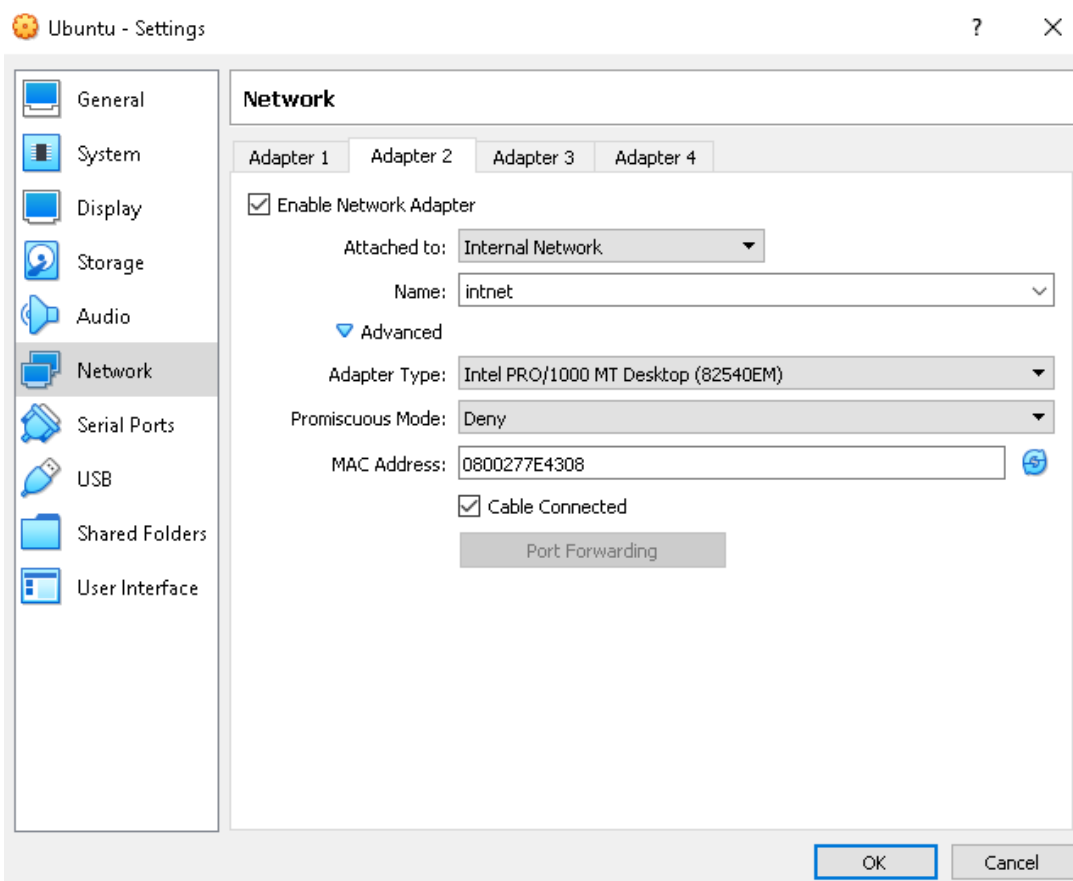
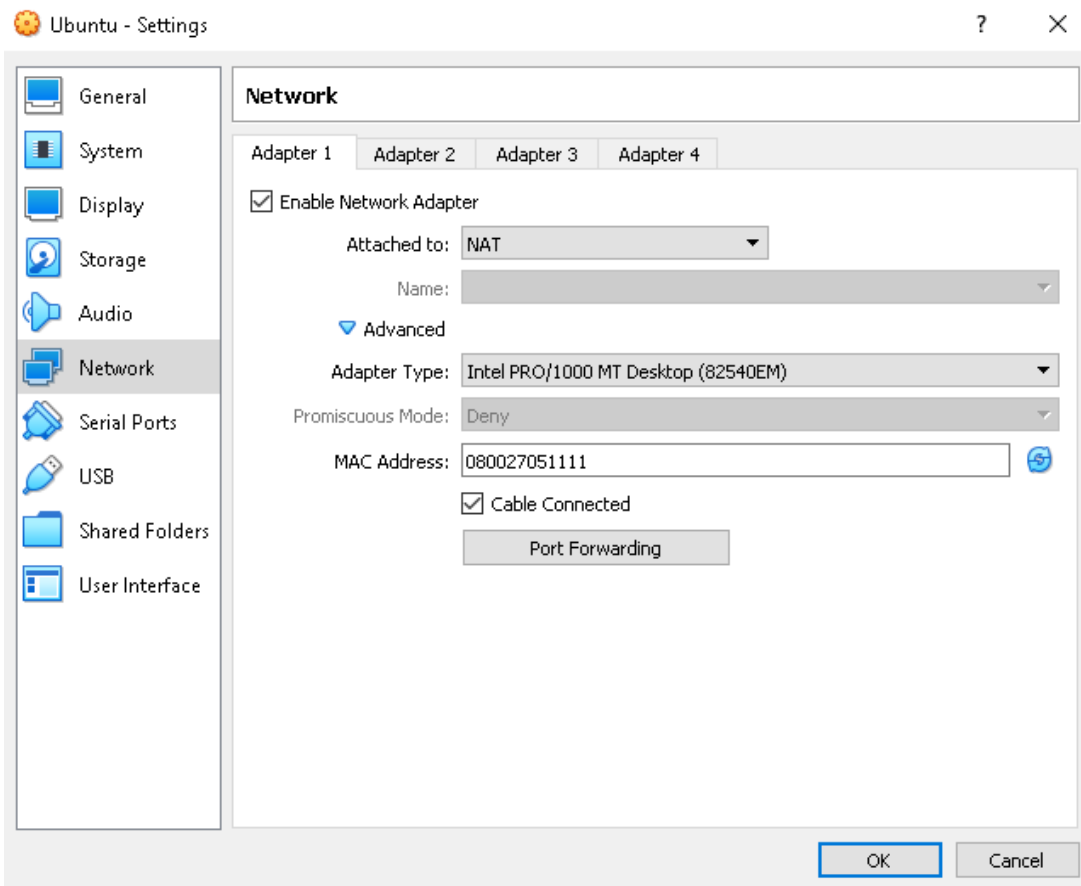
Confirm passphrase:

(X) Custom storage layout

Configuramos el particionamiento como nos pide la letra.



## Configuración de red en ubuntu



Creamos el usuario ansible

**useradd ansible**

creamos el home del usuario

**useradd ansible -m**

Creamos el password del usuario

**passwd ansible**

```
ansible@ubuntu1:~$ pwd
/home/ansible
ansible@ubuntu1:~$ hostname
ubuntu1
```

Vamos a ver las particiones creadas con LVM.

```
Disk /dev/sda: 15 GiB, 16106127360 bytes, 31457280 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x37d98c0e

Device      Boot  Start      End  Sectors  Size Id Type
/dev/sda1   *      2048    2099199    2097152    1G 83 Linux
/dev/sda2           2099200 31457279 29358080   14G 8e Linux LVM

Disk /dev/mapper/cl-root: 5 GiB, 5368709120 bytes, 10485760 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk /dev/mapper/cl-swap: 2 GiB, 2143289344 bytes, 4186112 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk /dev/mapper/cl-home: 3 GiB, 3221225472 bytes, 6291456 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes

Disk /dev/mapper/cl-var: 4 GiB, 4294967296 bytes, 8388608 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
```



Vamos a configurar la IP estática en Ubuntu.

Entramos en `/etc/netplan/00-installer-config.yaml` y agregamos la IP y el gateway de nuestro controlador ubuntu .

```
ansible@ubuntu1:~$ sudo vi /etc/netplan/00-installer-config.yaml
[sudo] password for ansible:
```

```
# This is the network config written by 'subiquity'
network:
  ethernets:
    enp0s3:
      dhcp4: no
      addresses: [192.168.43.42/24]
      gateway4: 192.168.43.1
      nameservers:
        addresses: [8.8.8.8,8.8.4.4]

  version: 2

|
~
```

Luego que configuramos la Ip estática con su gateway y dns aplicamos los cambios.

```
ansible@ubuntu1:~$ sudo netplan apply
ansible@ubuntu1:~$ |
```

Verificamos la ip en la interface enp0s3.

```
ansible@ubuntu1:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UN
KNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq
_codel state UP group default qlen 1000
    link/ether 08:00:27:63:8e:43 brd ff:ff:ff:ff:ff:ff
    inet 192.168.43.42/24 brd 192.168.43.255 scope global enp0
s3
        valid_lft forever preferred_lft forever
    inet6 2800:ac:8004:a880:a00:27ff:fe63:8e43/64 scope global
dynamic mngtmpaddr noprefixroute
        valid_lft 3546sec preferred_lft 3546sec
    inet6 fe80::a00:27ff:fe63:8e43/64 scope link
        valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOW
N group default qlen 1000
    link/ether 08:00:27:2b:71:d2 brd ff:ff:ff:ff:ff:ff
```

Vamos a configurar los nombre de hosts en el servidor Ubuntu

```
ansible@ubuntu1:~$ pwd
/home/ansible
ansible@ubuntu1:~$ hostname
ubuntu1
ansible@ubuntu1:~$ sudo vi /etc/hosts
[sudo] password for ansible:
ansible@ubuntu1:~$ |
```

Ingresamos las direcciones IP de los servidores y su nombre a sí podemos conectarnos entre ellos con solo el nombre de hosts

```
127.0.0.1 localhost
#127.0.1.1 ubuntu1

#IP hosts servidores

#192.168.0.143 centos
#192.168.0.142 ubuntu
#192.168.0.141 controlador

192.168.43.42 ubuntu
192.168.43.43 centos
192.168.43.38 controlador

# The following lines are desirable for IPv6 capable hosts
```

Chequeamos que podemos llegar a los distintos servidores con su nombre de host.

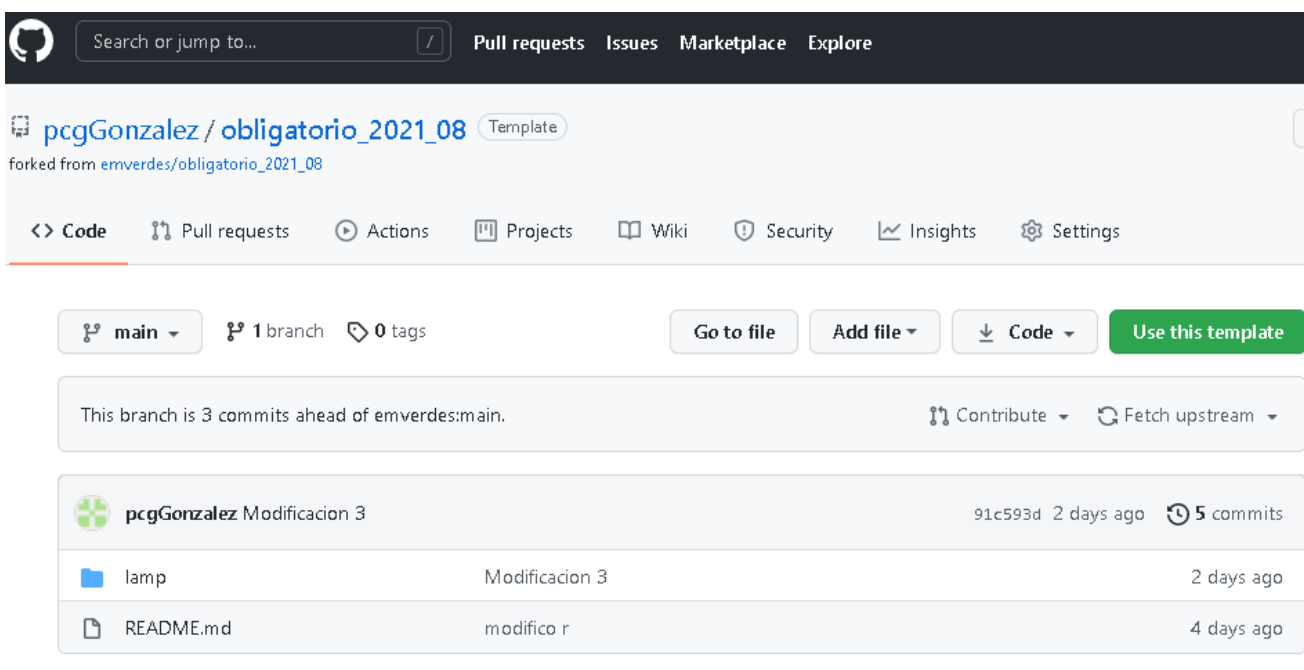
```
ansible@ubuntu1:~$ hostname
ubuntu1
ansible@ubuntu1:~$ ping centos
PING centos (192.168.43.43) 56(84) bytes of data.
64 bytes from centos (192.168.43.43): icmp_seq=1 ttl=64 time=1
.02 ms
64 bytes from centos (192.168.43.43): icmp_seq=2 ttl=64 time=1
.32 ms
^C
--- centos ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1002ms
rtt min/avg/max/mdev = 1.023/1.171/1.319/0.148 ms
ansible@ubuntu1:~$ ping controlador
PING controlador (192.168.43.38) 56(84) bytes of data.
64 bytes from controlador (192.168.43.38): icmp_seq=1 ttl=128
time=1.18 ms
^C
--- controlador ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 1.183/1.183/1.183/0.000 ms
ansible@ubuntu1:~$ |
```

En el equipo controlador vamos a instalar Visual studio code para tener un mejor manejo de nuestro repositorio github el mismo lo vamos a configurar con WSL para conectarnos a nuestro equipo remoto , y le vamos a instalar extensiones de ansible para que nos ayude a la hora de manipular código.

```

WLS TARGETS
  Ubuntu-20.04 default distro
    workspace.code-workspace /home/pablog/repo/obligatorio_2021_08/lamp/workspace.code...
    obligatorio_2021_08 /home/pablog/.ansible/obligatorio_2021_08
  
```

1) Realizamos el Fork del repositorio [https://github.com/emverdes/obligatorio\\_2021\\_08](https://github.com/emverdes/obligatorio_2021_08)



Se clonó el repositorio en el equipo local que será nuestro equipo “controlador”

```

pablog@PCGonzalez:~$ git clone https://github.com/pcgGonzalez/obligatorio_2021_08
Cloning into 'obligatorio_2021_08'...
remote: Enumerating objects: 82, done.
remote: Counting objects: 100% (16/16), done.
remote: Compressing objects: 100% (13/13), done.
remote: Total 82 (delta 3), reused 3 (delta 3), pack-reused 66
Unpacking objects: 100% (82/82), 17.85 KiB | 537.00 KiB/s, done.
pablog@PCGonzalez:~$
  
```

## 2) Corrija los errores que haya en los playbooks y también agregamos ansible.cfg

Vamos a agregar un archivo ansible.cfg y un archivo inventario.

Inventario.ini y playbooks.yml

```
pablog@controlador:~/repo/obligatorio_2021_08$ ls
README.md  lamp
pablog@controlador:~/repo/obligatorio_2021_08$ git init
Reinitialized existing Git repository in /home/pablog/repo/obligatorio_2021_08/.git/
pablog@controlador:~/repo/obligatorio_2021_08$ git add .
pablog@controlador:~/repo/obligatorio_2021_08$ git commit -m "Subo Inventario.ini "
[main dc0d561] Subo Inventario.ini
 1 file changed, 15 insertions(+)
pablog@controlador:~/repo/obligatorio_2021_08$ git push
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 4 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 453 bytes | 453.00 KiB/s, done.
Total 4 (delta 1), reused 0 (delta 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/pcgGonzalez/obligatorio_2021_08
 7255246..dc0d561  main -> main
pablog@controlador:~/repo/obligatorio_2021_08$
```

ansible.cfg habilitamos **remote\_user = ansible** , y le agregamos el usuario ansible.



Ahí le ponemos la ruta donde se encuentra nuestro inventario

```
[defaults]
inventory = ./home/pablog/repo/obligatorio_2021_08/lamp/hosts.ini
#library = ~/.ansible/plugins/modules:/usr/share/ansible/plugins/modules
#module_utils = ~/.ansible/plugins/module_utils:/usr/share/ansible/plugins/module_utils
#remote_tmp = ~/.ansible/tmp

2  #ansible_user=ansible
3  #ansible_password=ansible01
4  #ansible_ssh_private_key_file=/home/ansible/.ssh/id_rsa
5  #ansible_become=true
6
7  [all]
8  centos
9  ubuntu
10
11 [redhat]
12 centos
13
14 [debian]
15 ubuntu
```

El archivo **hosts.ini** tendrá listado de servidores remotos donde instalaremos los paquetes con opción de elegir en cuál distribución instalarlos .

El inventario venía definido con [webservers] y [dbservers] los cambiamos por [redhat] para los servidores que pertenecen a centos y por [debian] para identificar los servidores con ubuntu.

Lo primero que detectamos en las tareas y cambiamos fue el usuario remoto **root** por el usuario **ansible**.

```
1  ---
2  # This playbook deploys the whole application stack in this site.
3
4  - name: apply common configuration to all nodes
5    hosts: all
6    remote_user: ansible
7    become: yes
8
9    roles:
10     - common
```

También agregamos become: yes para permitir al usuario ansible tener privilegios elevados.

El el role common se quitaron los tags Se cambia el servicio NTP por Chrony

```
- name: Configure Chrony file
  template:
    src: "/home/ansible/repo/obligatorio_2021_08/lamp/roles/common/templates/chrony.conf.j2"
    dest: "/etc/chrony.conf"
    mode: 0600
  tags: chrony
  notify: restart chrony

# Iniciar servicio de chronyd
- name: Start the chrony service
  service:
    name: chronyd
    state: started
    enabled: yes
  tags: chrony
```

Para Debian se optó por instalar MySQL mientras que en RedHat se utilizó MariaDB Se agrega el componente pip para instalar las dependencias necesarias para la conexión entre python, php

```
# Se instala los paquetes en CentOS
- name: Install MariaDB package for CentOS
  yum:
    name: "{{ paquetes }}"
    state: present
  vars:
    paquetes:
      - mariadb-server

# Se instalan paquetes de python
      - python3-pip
      - python-mysqldb
  when: ansible_os_family == "RedHat"

# Se instalan los paquetes para Debian
- name: Install MariaDB package for Debian
  apt:
    name: "{{ paquetes }}"
    state: present
  vars:
    paquetes:
      - mariadb-server
      - python3-mysqldb

  when: ansible_os_family == "Debian"

# Configuración de Selinux
```

```
# Este manual despliega toda la pila de aplicaciones en este sitio.
```

```
- name: apply common configuration to all nodes
  hosts: all
  remote_user: ansible
  become: yes
```

```
roles:
| - common
```

```
- name: configure and deploy the webservers and application code
  hosts: all
  remote_user: ansible
  become: yes
```

```
roles:
| - web
```

```
- name: deploy MySQL and configure the databases
  hosts: all
  remote_user: ansible
  become: yes
```

```
roles:
| - db
```

## Errores al ejecuta playbooks

```
ansible@PCGonzalez:~/repo/obligatorio_2021_08/lamp$ ansible-playbook -i inventario.ini playbooks.yml

PLAY [apply common configuration to all nodes] *****

TASK [Gathering Facts] *****
fatal: [ubuntu]: FAILED! => {"msg": "Missing sudo password"}
fatal: [centos]: FAILED! => {"msg": "Missing sudo password"}

PLAY RECAP *****
centos      : ok=0    changed=0    unreachable=0    failed=1    skipped=0    rescued=0    ignored=
0
ubuntu     : ok=0    changed=0    unreachable=0    failed=1    skipped=0    rescued=0    ignored=
0

ansible@PCGonzalez:~/repo/obligatorio_2021_08/lamp$
```

```
PLAY [apply common configuration to all nodes] *****
skipping: no hosts matched

PLAY [configure and deploy the webservers and application code] *****
skipping: no hosts matched

PLAY [deploy MySQL and configure the databases] *****
skipping: no hosts matched

PLAY RECAP *****

ansible@PCGonzalez:~/repo/obligatorio_2021_08/lamp$
```

```
$ git init
error: could not lock config file /home/pablog/repo/obligatorio_2021_08/
fatal: could not set 'core.repositoryformatversion' to '0'
$ git add .
fatal: Unable to create '/home/pablog/repo/obligatorio_2021_08/lamp/.git
$ git commit -m " Subo Readme "
fatal: Unable to create '/home/pablog/repo/obligatorio_2021_08/lamp/.git
$ sudo git commit -m "subo Readme"
>>> /etc/sudoers: syntax error near line 21 <<<
sudo: parse error in /etc/sudoers near line 21
sudo: no valid sudoers sources found, quitting
sudo: unable to initialize policy plugin
```